

## Oficiální zadání BP



České vysoké učení technické v Praze  
Fakulta elektrotechnická



**ČVUT FEL Katedra počítačové grafiky a interakce**

Bakalářská práce

**Rozšíření 3D Studia Max pro renderování  
vícepohledových stereoskopických animací**

*Tomáš Buk*

Vedoucí práce: Ing. Zdeněk Trávníček

Studijní program: Softwarové technologie a management strukturovaný bakalářský

Obor: Web a multimedia

Květen 2010



## **Poděkování**

Rád bych poděkoval všem přátelům a známým za čas, který mám tu možnost trávit v jejich přítomnosti.

Děkuji panu Ing. Zdeňku Trávníčkovi za veškeré podněty a rady, kterými mě po celou dobu podporoval i za trpělivost, kterou měl.

Děkuji Zuzance, za všechny úsměvy a naděje, kterými mě obdařila. Svou přítomností vytvořila inspiraci a svými činy podmínky pro realizaci celého projektu.

Děkuji své rodině za neustálou podporu, které se mi dostává a které si nesmírně vážím.



## **Prohlášení**

Prohlašuji, že jsem svou bakalářskou práci vypracoval samostatně a použil jsem pouze podklady uvedené v příloženém seznamu.

Nemám závažný důvod proti užití tohoto školního díla ve smyslu §60 Zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon).

V Praze dne 20. 5. 2010

.....





## **Abstract**

This work concentrates primarily on stereoscopy. There is explained the concept itself here as well as its application and various technologies, which make the spatial experience possible. Furthermore, it covers the application of newly gained information on existing software, in order to accomplish the plugin creation. The existing program extensions are analysed and the reason to create a new module is sustained. Its design, implementation and final software overview is provided. The output is a plugin, which allows a variety of processes, connected with the creation of a stereoscopic projection, to be automated.

## **Abstrakt**

Hlavním zaměřením této práce je stereoskopie. Je zde vysvětlen samotný pojem, jeho aplikace a různé technologie, které prostorový zážitek umožňují. Dále je pokryta zakomponování nových poznatků do existujícího programového vybavení, za účelem vzniku zásuvného modulu. Jsou rozebrána stávající programová rozšíření, obhajují vznik nového pluginu. Je popsán jeho postupný návrh, implementace i zhodnocení finálního produktu. Výsledkem je zásuvný modul, který dovoluje mnoho procesů, spojených s více pohledovou stereoskopickou projekcí, zautomatizovat.



# Obsah

<b>1 Úvod .....</b>	<b>1</b>
1.1 Stereoskopie.....	1
1.2 Vymezení cílů práce.....	1
1.3 Používané pojmy.....	1
<b>2 Stereoskopie .....</b>	<b>3</b>
2.1 Princip prostorového vidění.....	3
2.2 Paralaxa.....	6
2.3 Možnosti záznamu.....	8
<b>3 Popis řešeného problému, vymezení požadavků na implementovaný systém .....</b>	<b>11</b>
3.1 Používané technologie.....	11
3.2 Popis řešeného problému - současný stav.....	15
3.3 Existující implementace.....	15
3.4 Požadavky na implementovaný systém.....	16
<b>4 Rozšiřovaný program .....</b>	<b>19</b>
4.1 3ds Max.....	19
4.2 Možnosti rozšíření 3ds Max.....	19
4.3 MaxScript.....	20
4.4 Požadavky na implementační řešení.....	21
4.5 Výsledný jazyk rozšíření.....	21
<b>5 Návrh .....</b>	<b>23</b>
5.1 Rozdělení podle uživatelských rolí.....	23
5.2 Model systému z hlediska architektury.....	23
5.3 Postup práce.....	24
5.4 Uživatelské rozhraní.....	25
<b>6 Implementace .....</b>	<b>27</b>
6.1 Vybrané funkce MaxScriptu.....	27
6.2 Napozicování odvozených kamer .....	28
6.3 Slepé uličky.....	28
<b>7 Testování .....</b>	<b>31</b>
7.1 Testování uživatelského rozhraní.....	31

7.2 Testování funkčnosti.....	33
<b>8 Závěr .....</b>	<b>35</b>
8.1 Zhodnocení dosažených cílů.....	35
8.2 Doporučení dalšího pokračování práce.....	35
<b>9 Použité zdroje .....</b>	<b>36</b>
<b>10 Obrazová příloha .....</b>	<b>38</b>
<b>Dodatek A - Obsah přiloženého CD .....</b>	<b>41</b>

# 1 Úvod

*Média na nás ze všech stran valí zahlcující množství informací, z nichž útržkovitě zaslechneme pojmy typu 3D, prostorový nebo stereo. V této části si osvětlíme nejen podstatu stereoskopie, tedy i těžiště této práce, ale budeme se zabývat i rozdílnou jazykovou interpretací.*

## 1.1 Stereoskopie

Pojem *stereo* pochází z řečtiny a lze jej přeložit jako „prostorový“, či „mající hloubku“. Stereoskopie ve své podstatě definuje jak pořizovat záznam kamerou či fotoaparátem tak, aby zachycená hloubka mohla být později adekvátně reprodukována.

Záznam prostorovosti okolního světa do vhodného stereoskopického formátu je jedna věc, jeho odpovídající prezentace věc druhá. V přirozeném prostředí získává člověk pojem o prostoru na základě mírně odlišných obrazů, pořízených levým a pravým okem. Tuto situaci je třeba simulovat i během reprodukce stereoskopického materiálu. Technologie, jaké se k prezentaci používají, jsou blíže specifikovány ve 3. kapitole.

## 1.2 Vymezení cílů práce

Výstupem této bakalářské práce je program, který lze jednoduše spustit z programu 3ds Max. Implementovaný modul je schopen na základě parametrů a naklíčovaných pozic zvolené referenční kamery spočítat a vytvořit kamery nové, umožňující snímat záběry pro zařízení typu CAVE (*Computer Aided Virtual Environment*) a autostereoskopické displeje. Samozřejmostí je generování klasických stereoskopických výstupů. Nově vytvořené kamery mají takové parametry, že jsou schopny renderovat nejen statické snímky, kdy kamera je po celou dobu v klidu, ale i scény, ve kterých se kamera pohybuje. Program je koncipován jako zásuvný modul (*plugin*) a je kladen důraz na jeho snadné a intuitivní ovládání, realizované pomocí grafického uživatelského rozhraní.

## 1.3 Používané pojmy

Informační technologie jsou právě jedno z těch odvětví lidské činnosti, kde výsostným a prakticky jediným používaným jazykem je angličtina. Ne ve všech případech je možné nalézt české ekvivalenty. Tato práce je však česky psaná a v opakujících se případech budu nadále používat výhradně českých názvů. Původní anglické pojmy uvedu kurzívou v závorce při prvním výskytu termínu v textu. Nepřeložitelné odborné názvosloví předložím

v originální, byť běžně používané podobě. Samozřejmostí je důkladné vysvětlení neznámých pojmů, se kterými budu v následujících sekcích operovat.

## 2 Stereoskopie

### 2.1 Princip prostorového vidění

#### 2.1.1 Monokulární klíče

Jedná se o vizuální ukazatele přijímané jedním okem. Vjem, kterým na mozek působí, vyvolá iluzi trojrozměrného obrazu.

Perspektiva (*perspective*) - lineární perspektiva zkresluje skutečnou velikost objektu v závislosti na jeho vzdálenosti od pozorovatele. Vzdušná perspektiva oproti lineární způsobuje, že objekty ve velké vzdálenosti mají menší kontrast a sytost barev kvůli atmosféře, která rozptyluje světlo. Daný jev také posouvá záření přijímané od vzdálených předmětů blíže k modré části spektra.

Akomodace (*accomodation*) - jedná se o změnu optické mohutnosti čočky, která přizpůsobí svůj tvar díky řasnatému tělísku. Člověk je tak schopen cíleně přestřovat ze vzdálených předmětů na blízké a naopak. Akomodace je potřeba jen u předmětů vzdálených do dvou metrů od pozorovatele. U vzdálenějších objektů zůstává optická mohutnost čočky téměř stejná. Ve vztahu ke stereoskopii je podstatné zmínit, že blízké předměty se jeví jako ostré, zatímco vzdálenější působí rozmazaně.

Okluze (*occlusion*) - úzce souvisí s lineární perspektivou. Jedná se o překrývání vzdálenějších předmětů těmi bližšími.

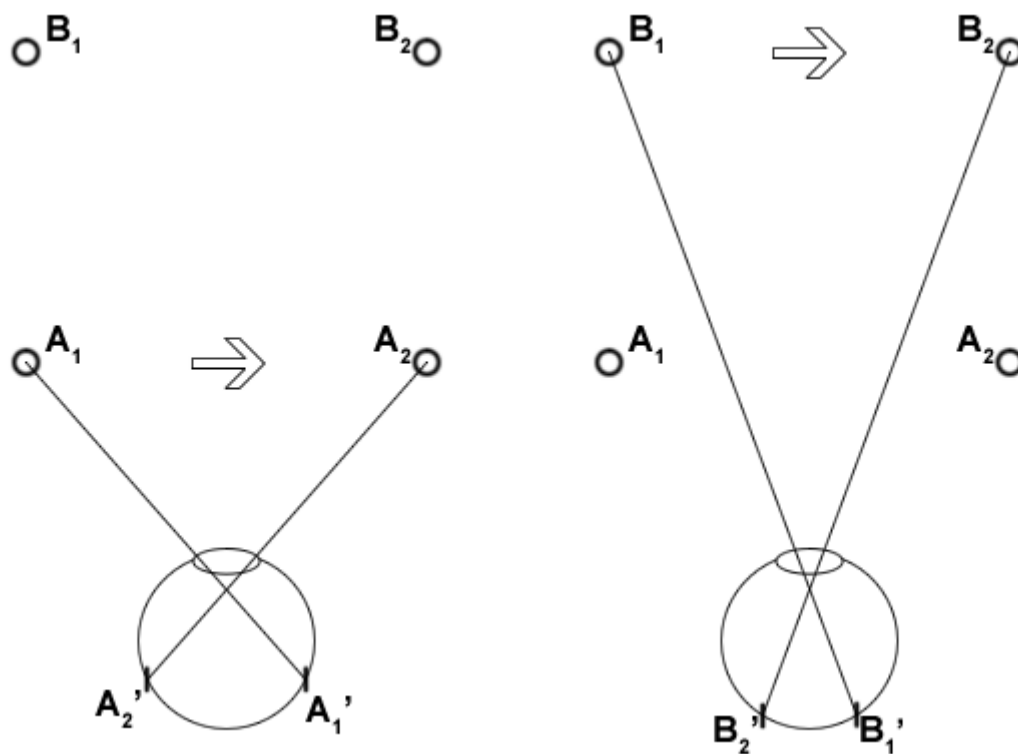
Periferní vidění (*peripheral vision*) - úhel pohledu, který se jeví jako zaostřený, se pohybuje kolem 45ti stupňů. Periferně jsme však schopni pozorovat úhle blízké 180ti stupňům. Rovné linie v periferních oblastech se jeví jako zakřivené. Názorněji je taková skutečnost viditelná na snímcích pořízených širokouhlým objektivem.

Gradient textur (*texture gradient*) - s rostoucí vzdáleností objektu se jednotlivé detaily povrchu postupně ztrácí a viditelná je pouhá aproximace jednotlivých odstínů. Zářným příkladem je dlážděná silnice. S rostoucí vzdáleností jsou spáry mezi jednotlivými dlaždicemi stále méně patrné.

Světlo a stín (*lighting and shading*) - způsob, jakým je předmět osvětlen, která místa jsou světlejší a která tmavší, dává informaci o tom, odkud světlo pochází, ale dává i ideu prostorového rozložení objektu. Analogicky vrhaný stín pomáhá určit, který předmět se nachází v popředí a který v pozadí.

Pohybová paralaxa (*motion parallax*) - (viz. Obr. 2.1) během přemístění pozorovatele

z jednoho bodu do druhého je zdánlivý pohyb stojících předmětů ovlivněn jejich vzdáleností od pozorovatele. Zatímco předměty v jeho blízkosti vyvolávají dojem stejně rychlého pohybu opačným směrem, objekty ve velké vzdálenosti jsou zdánlivě pořád v klidu



Obr. 2.1: Princip pohybové paralaxy. Během pohybu se krajní polohy bližšího předmětu na obraze sítnice promítnou s větší vzájemnou vzdáleností, než odlehlější předměty o stejných parametrech.

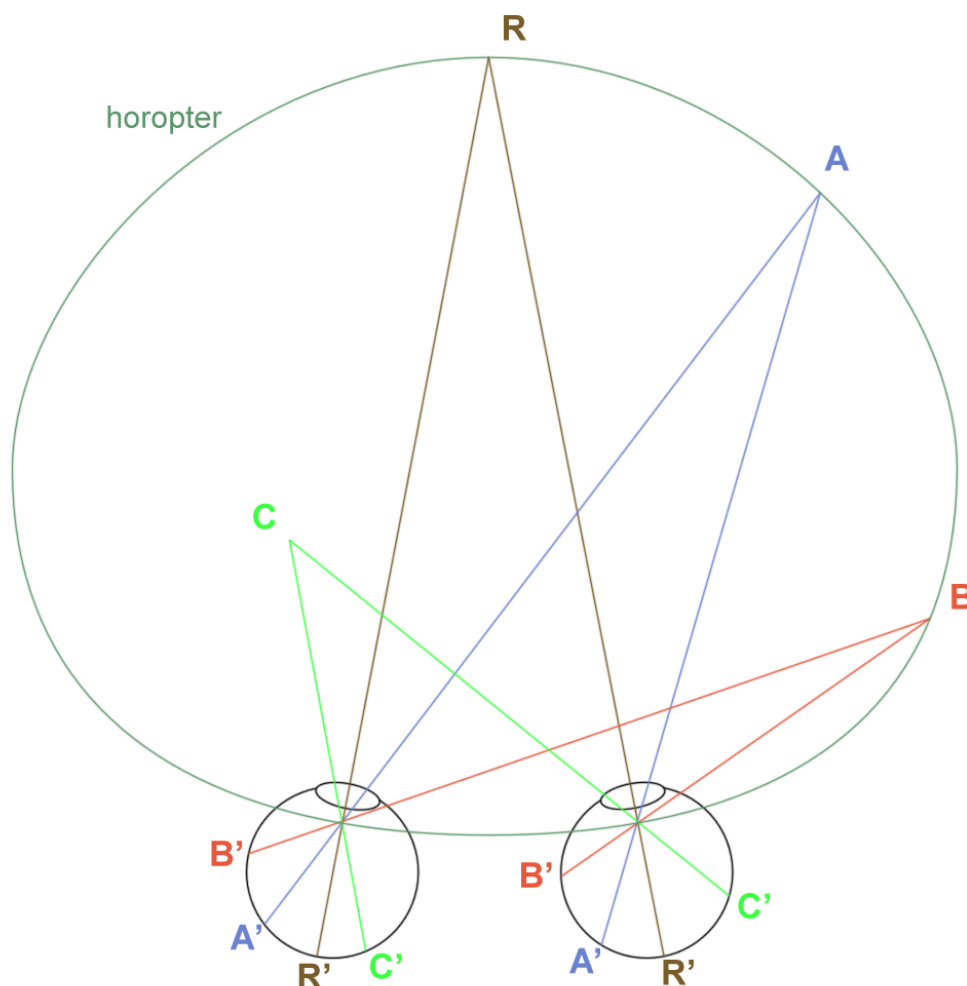
### 2.1.2 Binokulární klíče

Na rozdíl od monokulárních, binokulární klíče poskytují informaci o trojrozměrnosti prostoru na základě obrazu získaného oběma očima.

Binokulární disparita (*binocular disparity, stereopsis*) - disparita je vzdálenost, o kterou se liší odpovídající si obrazy pozorovaného bodu na sítnicích. Podstatným pojmem je tzv. horopter, což je uzavřená křivka, tvořená množinou bodů o nulové disparitě. U předmětů umístěných v pozici horopteru mají obrazy na sítnici shodný rozestup a pozorovateli se zdají být stejně vzdáleny. Na (Obr. 2.2) oči konvergují do bodu R a jeho obraz je promítnán do středu sítnice R'. Body A a B jsou oba umístěny na horopteru, tudíž odpovídající obrazy jsou zachyceny na sítnici s nulovou disparitou. Naproti tomu bod C leží mimo oblast



horopteru a z obrázku je patrné, že obraz C' má u každého oka jinou vzdálenost od bodu R', tedy odlišnou disparitu. Výsledného prostorového vjemu je dosaženo v mozku, který je, na základě odlišných pozorovacích úhlů pro každé oko, schopen triangulovat hloubku předmětu.



Obr. 2.2: Znárodněný jev se nazývá binokulární disparita. Předměty, jevící se ve stejné vzdálenosti leží na horopterové kružnici (o přesnou kružnici se nejedná).

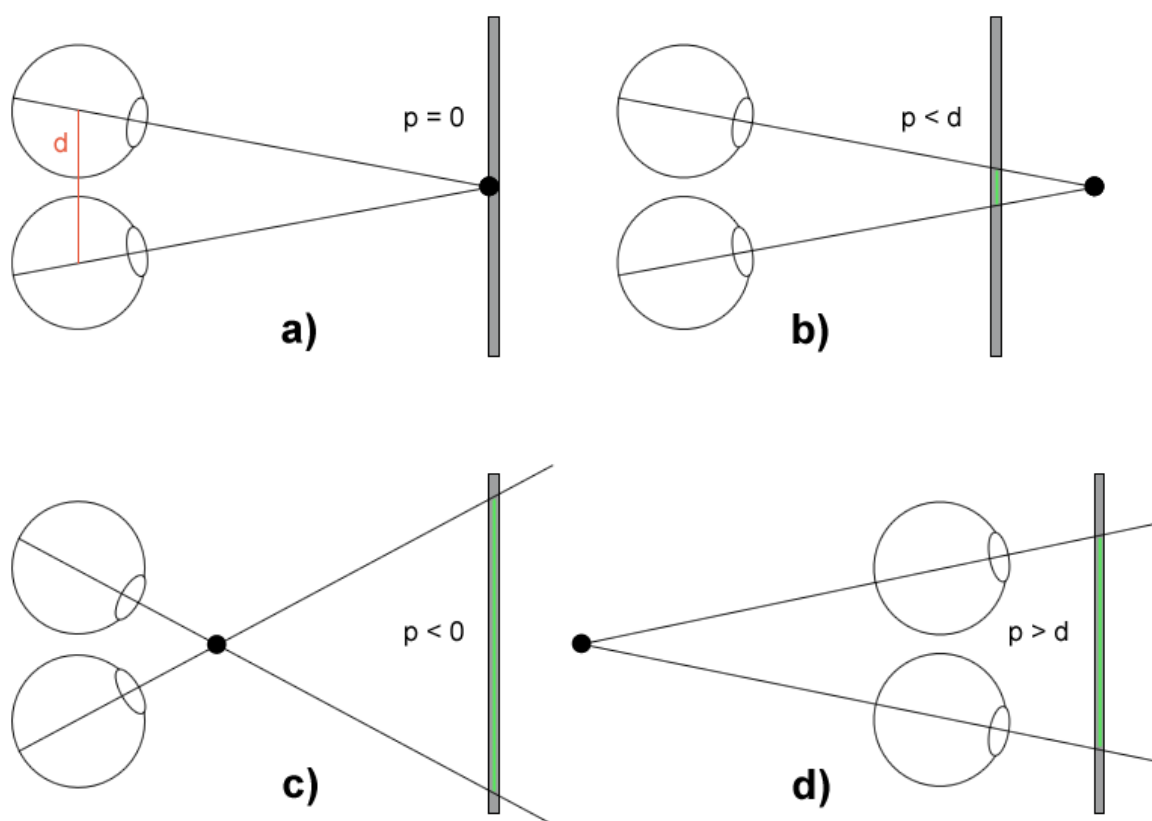
Konvergence (*convergence*) - základem binokulární disparity je schopnost očí se v jeden okamžik soustředit na stejný bod v prostoru, tedy konvergovat. Oční bulvy mají schopnost otáčet se kolem své vertikální osy a směřovat přijímané obrazy do středu sítnic. Děje se tak napínáním extraokulárních svalů.

### 2.1.3 Odvozené klíče

V některých publikacích se uvádí ještě tzv. odvozené klíče, které kombinují klíče monokulární a binokulární. Předpokladem jejich existence je vytvoření mentálního obrazu vnímaného prostoru. Studium existence odvozených klíčů však nepokročilo natolik, abych zde mohl uvést konkrétní fakta.

## 2.2 Paralaxa

Na dříve zmíněném principu binokulární dispartity je postaven další pojem, a tím je paralaxa. Jedná se o vzdálenost odpovídajících obrazů pravého a levého oka. V následujících odstavcích jsou podrobněji rozebrány různé případy paralax a jejich vlivu na výsledný stereoskopický efekt.



Obr. 2.3: a) nulová paralaxa, oči konvergují do roviny promítání b) pozitivní konvergentní paralaxa c) negativní paralaxa d) pozitivní divergentní paralaxa.

### **2.2.1 Nulová paralaxa**

(Obr. 2.3 a) Tento druh paralaxy nastává, pokud se optické osy očí protínají na promítací rovině. Ve výsledku to znamená, že obrazy obou sítnic jsou shodné. Mozek v tomto případě dostává dvě identické informace a potenciální prostorovost je schopen určovat pouze z monokulárních klíčů.

### **2.2.2 Pozitivní konvergentní paralaxa**

(Obr. 2.3 b) V případě, že zaostřený objekt se nachází až za rovinou promítání, hovoříme o pozitivní (konvergentní) paralaxe. V praxi stereoskopické projekce je tato paralaxa doporučována, protože klade nejmenší nápor na oči pozorovatele.

### **2.2.3 Pozitivní divergentní paralaxa**

(Obr. 2.3 d) V reálné situaci se s tímto druhem paralaxy nesečkáme. Znamenalo by to totiž, že optické osy se protnou až za snímacím zařízením, potažmo za hlavou diváka. K tomu lidský mozek uzpůsoben není a při tvorbě stereoskopického materiálu je třeba se uvedenému jevu vyvarovat.

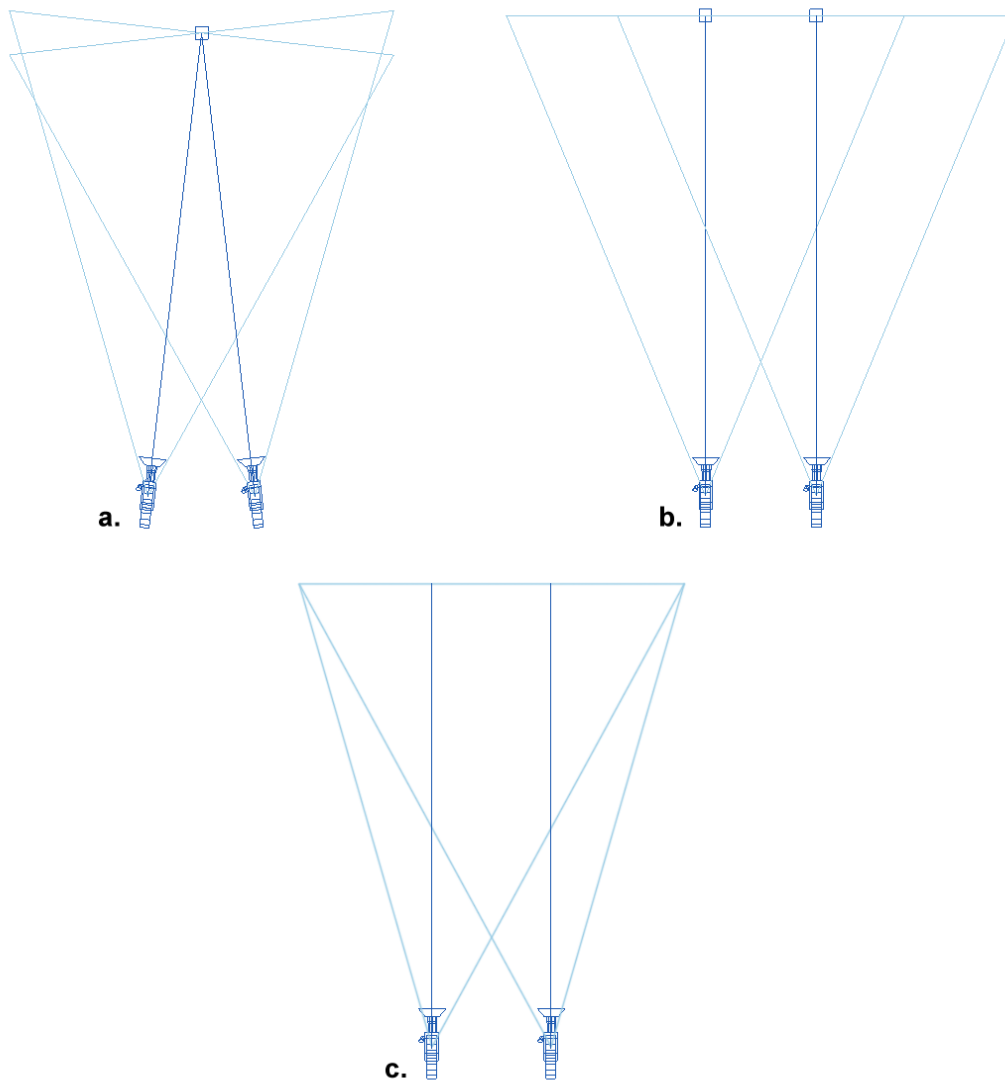
### **2.2.4 Negativní paralaxa**

(Obr. 2.3 c) Obrázek znázorňuje negativní paralaxu, která má za následek, že pozorovaný předmět je situován před promítací plochu. Přestože použitím negativní paralaxy je možné do snímku vnést silnější stereoskopický zážitek, její časté použití vede k brzké únavě očí, ve snaze neustále spojovat odpovídající obrazy. Také je třeba brát ohledy na promítací plochu, která způsobuje nepřirozený ořez vystupujících předmětů. Lidské oko je schopno lépe se vyrovnat s ořezem horizontálního okraje plátna, než vertikálního.

### **2.2.5 Vertikální paralaxa**

Pokud jsou ve scéně použity kamery sdílející společný cíl (viz. dále v této kapitole), do obrazu se dostane rušivý prvek zvaný vertikální paralaxa. Jde o příčný rozdíl odpovídajících si obrazů levého a pravého oka. Mozek se s tímto jevem nedokáže korektně vyrovnat a u výsledného snímku je často složité najít korektní hloubku.

## 2.3 Možnosti záznamu



Obr. 2.4: a) Konvergující kamery b) Paralelní kamery c) Kamery off-axis.

### 2.3.1 Konvergující kamery

Nejjednodušší nastavení kamer je znázorněno na (Obr. 2.4 a). Dvojice kamer je zde zaměřena na společný cíl a zdánlivě snímají stejnou plochu. Ve skutečnosti se nejedná o korektní záznam stereoskopického materiálu, protože zejména ve větších vzdálenostech

od středu se vyskytuje rostoucí vertikální paralaxa, uvedená v předešlém odstavci.

### **2.3.2 Paralelní kamery**

(Obr. 2.4 b) V praxi nejpoužívanější uspořádání kamer pro stereoskopické účely je paralelní. Kamery jsou od sebe vzdáleny stejně jako cíle, tudíž snímané obrazy leží v jedné rovině. Snímky levé a pravé kamery však nejsou shodné a před reprodukcí takto pořízeného obrazu je nutné přebytečné okraje ořezat.

Speciálním případem paralelních kamer jsou kamery *off-axis* (Obr. 2.4 c). Těží z výhod paralelního uspořádání, ale na rozdíl od něj mají nesoustřednou osu snímání a není proto zapotřebí dodatečný ořez záběru. Bohužel 3ds Max tento typ kamer v době psaní této práce nepodporuje a jediným způsobem, jak je utvořit, je použít externí plugin SVI Stereo 3D Renderer (viz. odstavec existující implementace).

### **2.3.3 Divergující kamery**

Kamery, jejichž cíle se protnou až za kamerou, jsou označovány jako divergentní. Reprodukovaný obraz obsahuje nejen pozitivní divergentní paralaxu, ale i paralaxu vertikální. V praxi se tento typ uspořádání nepoužívá a uvádím jej zde pouze pro úplnost.



### **3 Popis řešeného problému, vymezení požadavků na implementovaný systém**

*V této kapitole se budeme věnovat detailnějšímu popisu používaných technologií. Dále si ukážeme, jak zadání této práce souvisí se zařízeními typu CAVE, či autostereoskopický displej. Za závěr kapitoly shrneme seznam požadavků na implementovaný systém.*

#### **3.1 Používané technologie**

##### **3.1.1 Metody stereoskopického zobrazení**

###### **3.1.1.1 Anaglyph**

Patrně nejjednodušší zobrazovací technikou je anaglyph. Její princip spočívá ve vyextrahování určité barevné informace z dílčího záběru. V aditivním barevném modelu RGB by to znamenalo pro jeden snímek zachovat pouze informaci o modrém a zeleném kanálu, druhý snímek je zobrazen použitím pouze červeného kanálu. Výsledný snímek vzniká proložení obou snímků nad sebe. Sledování složeného obrazu pouhým okem ještě k žádnému stereoskopickému zážitku nevede. Je nutné použít brýle opatřené barevnými skly. Norma diktuje aplikaci červeného pro levé oko a azurového (modrá a červené) pro pravé. Pohled přes červený filtr nám umožní vnímat červenou barvu jako bílou a azurovou jako barvu odpovídající černé. Teprve tehdy získává mozek mírně odlišné obrazy totožné scény a je schopen vyprodukovat iluzi prostoru.

###### **3.1.1.2 Pasivní projekce**

Jednou z charakteristik světla jako elektromagnetického vlnění je, že vektor intenzity elektrického pole je kolmý na směr šíření. Pokud je světlo lineárně polarizované, vektor intenzity elektrického pole kmitá stále v jedné přímce. Dále může být světlo polarizováno kruhově, nebo elipticky.

Pasivní projekce využívá vždy jedné ze zmíněných polarizací světla. Zobrazovací technika se většinou skládá z dvojice projektorů a to buď vertikálně, nebo horizontálně umístěných. Před každým z projektorů je umístěn polarizační filtr, který zajišťuje navzájem opačnou polarizaci promítaných obrazů. Je použito speciální plátno, které polarizaci zachovává. Pozorovatel má k dispozici brýle, které díky stejným polarizačním filtrům, jakých je použito u projektorů, propouštějí pro každé oko mírně odlišné obrazy. Pokud je scéna sledována bez brýlí, lidské oko nerozezná rozdíl v polarizaci a obraz se jeví velmi rozostřený.

### **3.1.1.3 Aktivní projekce**

Aktivní projekce využívá jediný projektor, který pracuje s dvojnásobnou frekvencí oproti běžnému. Na klasickou promítací plochu je střídavě zobrazován snímek pro levé i pravé oko. Celková frekvence je 120Hz, každému oku tak připadne 60Hz. Stejně jako v případě pasivní projekce, pouhým okem se obraz zdá být rozmazaný, proto i zde je zapotřebí speciálních brýlí. Jsou však synchronizovány s projektorem a v momentě, kdy je promítán obraz pro levé oko, clona před pravým okem se zatemní. Analogicky se ztmaví clona před levým okem, pokud právě promítaný obraz je určen pro pravé oko. Synchronizační signál je z projektoru do brýlí veden pomocí kabelu, nebo mobilněji užitím radiových vln.

### **3.1.1.4 Infitec**

Pilířem této metody, vyvíjené německou firmou INFITEC GmbH, je proměnná citlivost oka na světlo různých vlnových délek. Za pomoci speciálního filtru lze dosáhnout posunutí vlnové délky světla k vyšší hodnotě, při zachování stejné barvy vnímané okem.

Jedná se v podstatě také o pasivní způsob projekce. Základem jsou dva synchronizované a zkalibrované projektory. Ty na stejnou plochu promítají v jeden okamžik stereoskopické snímky popřípadě video. Rozdíl je pouze v tom, že projektory mají navzájem posunuté hranice vlnových délek jednotlivých kanálů promítaného obrazu. K jejich správné interpretaci pro každé oko je zapotřebí brýlí, které posouvají vlnovou délku odpovídajícího obrazu na původní hodnotu.

### **3.1.1.5 Porovnání jednotlivých zobrazovacích technik**

#### **Anaglyph**

##### **Výhody**

- + Cena
- + Nenáročné vybavení
- + Možnost vytisknout stereoskopický obraz

##### **Nevýhody**

- Věrnost barev
- Neschopnost zobrazit všechny barevné odstíny



## **Pasivní stereoprojekce**

### **Výhody**

- + Zachovává 70 - 80 % světelnosti
- + Cenově dostupné brýle
- + U kruhově a elipticky polarizovaných odpadá nutnost pozorovat vzpřímeně

### **Nevýhody**

- Artefakty zobrazení
- Potřeba dvou synchronizovaných a kalibrovaných projektorů
- Speciální promítací plocha, zachovávající polarizaci

## **Aktivní stereoprojekce**

### **Výhody**

- + Promítání na libovolné plátno
- + Použití pouze jednoho projektoru
- + Nízká úroveň zobrazovaných artefaktů

### **Nevýhody**

- Zachovává pouze 15 - 20% světelnosti
- U některých pozorovatelů bolest hlavy způsobená rychlým blikáním
- Nákladné brýle synchronizované s projektorem

## **Infitec**

### **Výhody**

- + Nenáročné brýle
- + Žádná synchronizace brýlí s projektorem
- + Výborná separace obrazu

- + Běžná promítací plocha

#### **Nevýhody**

- Barevné odchytky
- Potřeba dvou kalibrovaných a synchronizovaných projektorů

### **3.1.2 CAVE**

(*Computer Aided Virtual Environment*) je zařízení, které pozorovateli umožní být zcela obklopen virtuální scénérií. Jedná se v podstatě o krychli či kvádr, jehož hrany jsou tvořeny kovovou konstrukcí. Každá ze stěn má vlastní projektor (nebo projektory, viz výše), který na její povrch promítá odpovídající obraz. Dle sofistikovanosti CAVE se běžně promítá na 3, 4, 5 nebo všech 6 stěn. Protože se zde uplatňuje stereoskopická projekce, pozorovatel je vybaven speciálními brýlemi a obrazy jednoho objektu na různých stěnách vytvářejí iluzi, že ten vystupuje do prostoru.

Zatímco pro přední, zadní a boční pohledy lze projektory umístit do přímého okolí CAVE, s pohledem horním a dolním by rozměry místnosti musely být poněkud větší. Značné úspory místa je docíleno použitím zrcadel, která s minimální ztrátovostí přenesou promítaný obraz na adekvátní stěny krychle, přičemž projektory mohou být umístěny více variabilně.

CAVE může být vybaven systémem pro trekování pozice pozorovatele. Děje se tak za pomoci elektromagnetických sensorů, které sledují pohyb pozorovatelových očí a podle jejich umístění v reálném čase upravují promítané snímky na jednotlivých stěnách. Nezbytností je, aby v takovém případě byla konstrukce CAVE zhotovena ze speciální nemagnetické oceli.

Není divu, že právě reálný dojem vystupujících předmětů má za následek rozšíření zařízení typu CAVE např. do automobilek (viz. Obr. 10.1), kde si designéři mohou „projít“ model právě navrhovaného vozidla a ujistit se o vhodnosti umístění ovládacích prvků či o celkovém působení interiéru vozidla.

### **3.1.3 Autostereoskopický displej**

Prozatím jediné zařízení, které je schopné poskytovat stereoskopický obraz bez speciálních brýlí, je autostereoskopický displej. Princip spočívá v mírně odlišném úhlu, pod kterým je záření zachyceno lidským okem. Autostereoskopický displej je většinou tvořen LCD, potažený speciální fólií - tzv. prizmová mřížka. Ta má za úkol lámat záření vycházející

z jednotlivých pixelů do různých směrů. Pokud je tedy autostereoskopický displej schopen lámat světlo do dvou směrů, pozorovatel před displejem vidí jedním okem lichý sloupec pixelů a druhým sudý. V praxi jsou autostereoskopické displeje schopny lámat světlo i pro osm různých úhlů pohledu, je tudíž možné umístit před displej více pozorovatelů a každý vidí scénu z nepatrně jiného úhlu. Přesto není autostereoskopický displej vhodný pro širší publikum kvůli vysoké pořizovací ceně a relativně malé velikosti plochy.

### **3.2 Popis řešeného problému - současný stav**

Nastavení programu 3ds Max nám ve své výchozí podobě neumožňuje vytvářet žádný druh stereoskopických kamer. Potenciální uživatel otevře scénu obsahující animaci a na základě již specifikované kamery chce vytvořit kamery nové, odpovídající jednotlivým pohledům. Jednou z možností je investice do většinou komerčních pluginů (viz. dále). Avšak ani takový zásuvný modul nám neumožňuje generovat více stereoskopických pohledů v závislosti na referenční kameře, vhodných pro prezentaci v CAVE. Naopak vytvářet větší množství kamer v rovině jednoho pohledu užitím externích modulů lze.

### **3.3 Existující implementace**

#### **3.3.1 Stereographer Max**

(<http://www.burningpixel.com/Max/StereoG.htm>)

Jedná se o starší freeware plugin, vytvořený v době čtvrtého vydání 3ds Max. Po vybrání zdrojové kamery umožňuje renderovat statický snímek nebo animaci a výsledek zkomponovat do jediného obrázku a to včetně anaglyphu. Nepodporuje výstup pro CAVE, ani autostereoskopický displej.

#### **3.3.2 SVI Stereo 3D Renderer**

(<http://www.spatialview.com/en/3dsmax>)

Profesionální řešení, vyvíjené společností SPATIAL VIEW. Uživatelský komfort je vskutku na úrovni a možnosti nastavení jsou široké. Zajímavostí je i dodatečná implementace Off - axis kamer. Bohužel renderovat lze pouze čelní pohled.

#### **3.3.3 3D Camera for 3ds Max 2.2.0**

(<http://www.visumotion.com/>)

Dalším z řad komerčních produktů je i zdánlivě primitivní 3D Camera. Oproti konkurenci

poskytuje výhodu ve formě generování dvou až devíti kamer, vycházejících z referenční kamery. Bohužel podpora vícehledového renderu i zde chybí.

### **3.4 Požadavky na implementovaný systém**

Předně je nutné stanovit, která kamera bude pokládána za referenční. Bez ohledu na vybranou kameru je možné zvolit, zda cílové zařízení bude CAVE, či autostereoskopický displej.

Volbou renderu pro CAVE uživatel blíže specifikuje stěny, pro které se budou utvářet jednotlivé kamery. Zvláště pro animátory je vhodné nechat zobrazit tzv. Stereo Safe Frame, ztělesněný dvěma rovinami. Blíže k referenční kameře se nachází rovina určující maximální přiblížení snímaného tělesa k objektivu, které je ještě pro lidské oko přijatelné. Druhá rovina určuje druhou mez, tedy kdy se snímané těleso jeví stále jako trojrozměrné. Objekty umístěné za touto vzdáleností se zdají být již v rovině.

Autostereoskopický displej umožňuje blíže specifikovat počet kamer, z jejichž záběrů se bude komponovat výsledný obraz. V závislosti na celkovém počtu kamer je třeba vždy nastavit relativní vzdálenost kamer, protože na rozdíl od klasické stereoskopie zde není platný obecný poměr mezi vzdáleností kamery od cíle a vzájemnou vzdáleností kamer.

Vyrenderované záběry je možné komponovat do různých podob. Nejrozšířenější je volba oddělených snímků (*separate images*), kdy každý vyrenderovaný snímek je uložen do zvláštního souboru. Další možností jsou horizontálně složené snímky (*horizontally composed*), kdy se nejprve vyrenderuje první snímek, poté druhý a je umístěn do stejného souboru jako první, ale nachází se pod ním. Obdobná je i volba vertikálního zarovnání, kde oba vyrenderované snímky jsou uloženy taktéž do jednoho souboru, ale v pořadí, kdy druhý snímek je umístěn napravo od prvního. Z rozličných důvodů užití je obsažena i funkce, která prohazuje pořadí ukládaných snímků (*swap images*).

Pro podporu ukládání videosekvencí je napodobena funkce time output z klasického menu Render Scene Dialog. První možností výstupu je statický snímek. Další je renderování animace s délkou trvání rovnou aktivnímu časovému segmentu zobrazenému na časové ose. Animace nemusí být renderována v celé délce, lze blíže specifikovat počáteční i koncový snímek.

Komplexní scéna, obsahující množství geometrie, velmi zatěžuje výpočetní jednotku a render testovacích scén je pak neúměrně zdlouhavý. Z tohoto důvodu je počítáno s možností nastavení výstupního rozlišení renderu.

Vyrenderovaný snímek je uložen do složky, kterou specifikuje, popř. ji sám vytvoří. Pro snazší orientaci v generovaných snímcích je možné vložit pro každý renderovaný projekt uživatelem definovanou předponu souboru. Pokud není nastavena, je použita výchozí. V případě renderu statického snímku má uživatel na výběr z několika výstupních formátů. Pokud se generuje animace, renderuje se automaticky do formátu .avi .

Před tím, než je započat samotný render, je možnost si jednotlivé kamery prohlédnout v pozici, v jaké se budou nacházet během renderu. Zejména pokud je generována animace, musí mít animátor přehled, zda trasa nově spočítaných kamer je správná a jestli není nutné upravit trasu té referenční. Render započne po zvolení stejnojmenné funkce. Postupně jsou renderovány pohledy ze všech kamer, je nastaveno jejich případné složení, velikost rozlišení a umístění výsledného souboru.



## 4 Rozšiřovaný program

Zaměřím se na pracovní prostředí programu 3ds Max, pro který je zásuvný modul zamýšlen. Popíši hlavní rysy tohoto, v profesionální grafické tvorbě hojně používaného programu, přičemž důraz bude kladen na získání základních rysů integrovaného skriptovacího jazyka. Následně zanalyzuji možnosti rozšíření.

### 4.1 3ds Max

3ds Max (dříve *3D Studio*, nebo *3D Studio Max*) je program vyvinutý společností Autodesk. Je určený pouze pro platformu Windows a to jak ve 32-bitové, tak i 64-bitové verzi. Jak již název napovídá, je určený primárně pro tvorbu 3D grafiky, vizualizace a animace. Používá se v herním průmyslu, při tvorbě speciálních filmových efektů, ale i v architektonických vizualizacích, při návrzích šperků či v průmyslovém designu.

Hlavní síla tohoto softwaru tkví v jeho modelovacích schopnostech. Zejména v oblasti herního průmyslu se výlučně používají modely složené z polygonů. 3ds Max podporuje širokou škálu modelovacích nástrojů použitelných na různých hierarchických úrovních polygonálního modelu, ať již jsou to koncové body, hrany, hranice, polygony nebo celý objekt. Od verze 2010 je pro polygonální modelování nový modul *Graphite*, který výrazně zefektivňuje proces tvorby i následné úpravy modelu, integrováním užitečných funkcí z pluginů od externích vývojářů. Kromě rozsáhlého polygonálního modelování lze pracovat také s plochami z křivek - NURBS, ale v tomto ohledu 3ds Max značně zaostává za programem Maya, vyvíjeným jinou divizí Autodesku.

Animaci je možno posunout o úroveň výše zapojením částicových systémů, vytvořit reálně se chovající části oděvu, či užít integrovaný fyzikální engine a zachytit reálně se řítící stěny budovy.

Při renderu máme na výběr ze dvou modulů - standardní *Scanline Renderer* a poměrně nový, robustní - *Mental Ray*. Scanline se již moc nepoužívá, v širším měřítku se s ním setkáme pouze u tzv. *Active Shade*, kdy jakákoliv změna objektu v modelovacím prostoru je ihned překreslena do přibližné podoby renderu, a to právě použitím Scanline Rendereru. Uživatelsky zajímavější je Mental Ray s množstvím přesných shaderů, podporou nepřímého osvětlení, kaustických efektů a mnohým dalšími parametry.

### 4.2 Možnosti rozšíření 3ds Max

Existují dva způsoby, jak rozšířit existující funkcionalitu programu. První způsob využívá

integrovaný skriptovací jazyk MaxScript. Mezi podstatné výhody patří jeho okamžitá použitelnost ihned po nainstalování samotného 3ds Max. Komfortnější psaní skriptu poskytuje rovněž obsažený editor jazyka, vybavený zvýrazňováním klíčových slov, pokročilými volbami formátování kódu, či automatickým doplňováním slov (*auto complete*). Detailnější soupis kladů a záporů je rozepsán v následujícím odstavci.

Druhou možností je využít C++ přes vestavěné SDK. Výhodami jsou zejména objektový přístup, využívající například polymorfismus, dědičnost, či zapouzdření. Oproti skriptovacím jazykům je kód zkompilovaný, což poskytuje nesporný benefit v podobě vyššího výkonu. Také je k dispozici široká škála knihoven, použitelných nejen při práci s grafikou, ale třeba i XML, ve kterém se často ukládají trojrozměrná data.

### **4.3 MaxScript**

MaxScript je jazyk používaný výlučně v programech 3ds Max a 3ds Max Design (což je odlehčená verze, určená především pro tvůrce architektonických vizualizací). Je dosažitelný ihned po spuštění programu a umožňuje vytvořit takový zásuvný modul, který je načten automaticky při každém startu.

Jedná se o částečně objektově orientovaný jazyk, umožňující díky mnoha předdefinovaným funkcím pohybovat se na vyšší úrovni abstrakce. Poskytuje kompletní přístup ke knihovnám primitivních objektů a materiálů, dovoluje použít lokální souřadný systém libovolného objektu, či pracovat užitím stejných postupů jako v klasickém uživatelském rozhraní programu. Pro usnadnění animace lze jediným povelům spustit mód automatického tvoření klíčů, nebo pracovat s reálnými objekty ve scéně při zachování stejné hierarchie.

Syntaxe MaxScriptu je v mnoha ohledech podobná jazyku C. Vzhledem k minimálnímu počtu formátovacích pravidel je snadný pro používání i uživatelům bez zkušeností s programováním. Naopak zkušenému programátorovi navyklému na silnou typovou kontrolu bude dělat potíže zvyknout si na často zcela neopodstatnělé chování skriptu. Například mechanismus odchylování výjimek obsažen je, bohužel bez možnosti zachytit konkrétní výjimku. Narozdíl od jazyka C je vybaven garbage collectorem, dlužno říci, že ne zrovna povedeným. I po řádném ukončení skriptu a jeho opětovném nahrání se zcela běžně stane, že neinicializované proměnné jsou záhadně inicializovány na hodnotu z předchozího spuštění. Pokud je ve fázi ladění skript spuštěn bez jediné chyby, vyplatí se nejen MaxScript Editor, ale rovnou celé 3ds Max restartovat a zkusit skript spustit nanovo. Nezřídka se totiž stává, že



po restartu se najednou objeví vyjímka (obvykle způsobená neinicializovanou proměnnou), která se v předchozím spuštění vůbec nevyskytla.

Přestože na matematicky náročnější výpočty obsahujícími mnoho cyklů a výpočtů je vhodnější použít kompilované C++, nekompilovaný MaxScript má tu výhodu, že mnoho funkcí spojených s vektory, maticemi a lineární algebrou je již naprogramovaných a odzkoušených. Spolehlivě tak lze přistupovat k rotaci objektu užitím kvaternionů, nebo provádět operace nad transformační maticí objektu.

Další výhodou MaxScriptu je Macro Recorder, který zaznamenává veškerou uživatelskou činnost a posílá ji formou zpráv na běžný výstup. Cyklicky prováděné úkony lze takto zaznamenat a následně vložit do nového skriptu, který je jediným tlačítkem spuštěn.

#### ***4.4 Požadavky na implementační řešení***

Implementovaný modul bude ovládán pomocí grafického uživatelského rozhraní a uživateli poskytne přehledné a jednoduché ovládání. Umístění samotných kamer bude kompletně realizováno skriptem, za pomoci trigonometrických funkcí a animačních controllerů. Potřebná přesnost dosahuje alespoň čtyř desetinných míst. U vyrenderovaných snímků je důraz kladen na operace na úrovni pixelů, které budou využívány u skládání snímků do jediného souboru. Z logiky věci vyplývá, že finální výstup renderu je třeba ukládat na pevný disk a uživatel bude mít možnost specifikovat danou cestu k souboru.

#### ***4.5 Výsledný jazyk rozšíření***

Na základě analýzy požadavků, uvedené v předchozím odstavci, vyšlo najevo, že nejlepší volbou pro realizaci pluginu je skriptovací jazyk MaxScript. Grafické uživatelské rozhraní lze vytvořit přidáním předpřipravených komponent nebo přímo ve WYSIWYG editoru zvaném Visual MaxScript. Podpora trigonometrických funkcí je v jazyce C++ samozřejmostí, ve skriptovací alternativě jsou obsaženy již od prvního vydání. Naopak animační controllery, bez kterých je téměř nemožné sledovat trasu původní kamery, v dřívějších verzích SDK chybí. Požadavek na přesnost čísel s plovoucí desetinnou čárkou lépe splňuje C++. MaxScript přesto disponuje 64bitovými proměnnými, ale implementovaný modul je zamýšlen k použití na všech distribucích 3ds Max od verze 9, tudíž i stávajících 32-bitových. Další ze standartních funkcí MaxScriptu umožňuje pracovat s bitmapou na úrovni pixelů pomocí funkcí `getPixels()` a `setPixels()`, kde v obou případech je pracováno s řádkem pixelů ve formě pole. C++ není takových operací schopno bez dodatečné knihovny. Posledním z požadavků je práce se soubory, což je v jazyce C++ zcela standartní rys a nechybí ani u jazyka MaxScript.



## **5 Návrh**

*V této části se budu věnovat návrhové části. Na základě shromážděných požadavků bude má snaha směřovat k efektivnímu a fungujícímu návrhu, který bude stavebním kamenem následné implementace.*

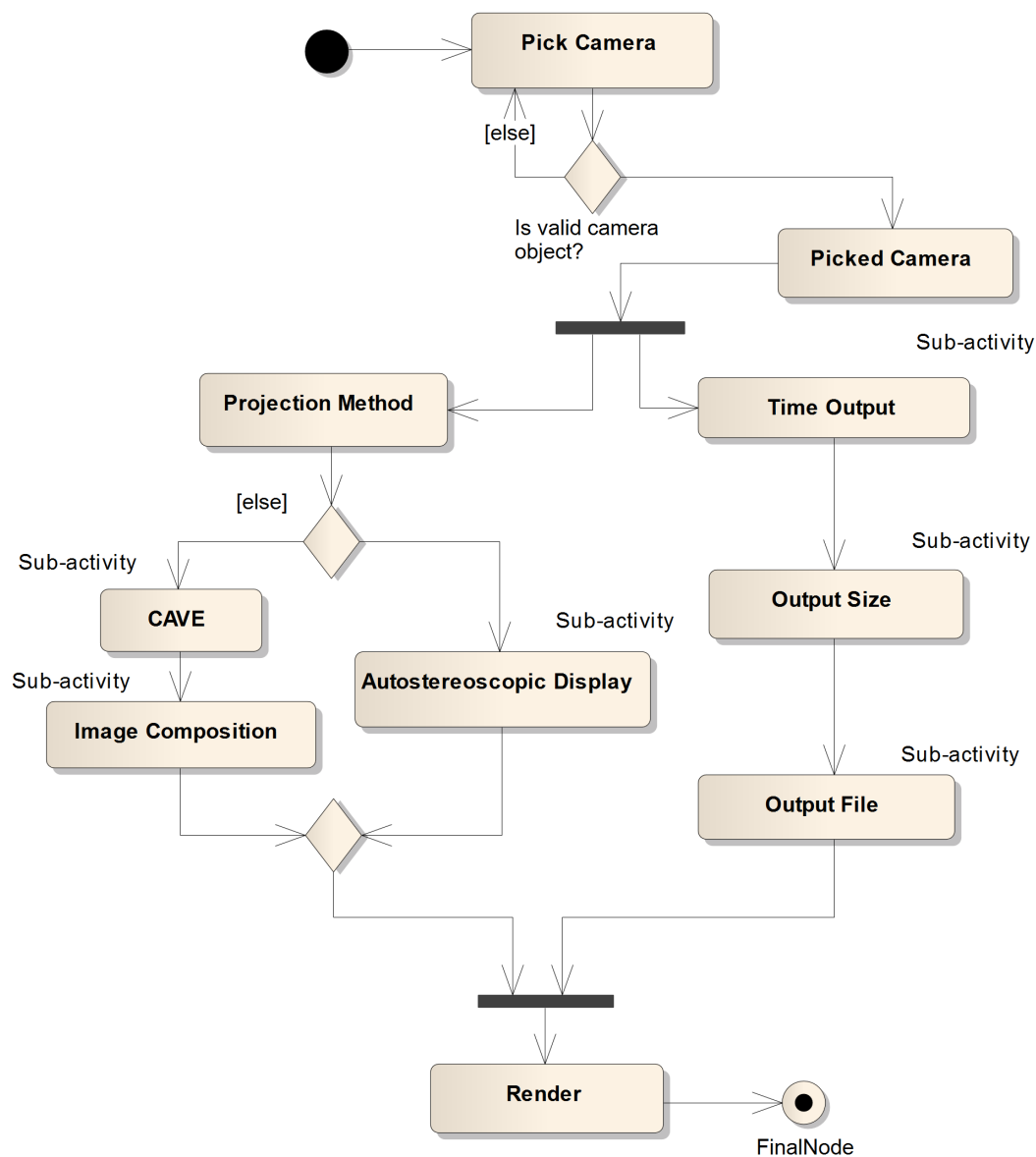
### **5.1 Rozdělení podle uživatelských rolí**

Zamýšlený modul je vyvíjen jako jednouzivatelský. Uživatelské role nejsou nikterak rozlišovány a není tedy počítáno ani s možnými restrikcemi stran funkčnosti.

### **5.2 Model systému z hlediska architektury**

Navrhovaný plugin je koncipován jako jednovrstvá aplikace. Kompletní funkčnost programu je tedy zahrnuta v jediné aplikaci. Důvod ke zvolení tohoto řešení, oproti používanějším vícevrstevným aplikacím, je jednak absence jakékoliv komunikace s databází a jednak fakt, že modul postrádá síťové rozhraní.

### 5.3 Postup práce



Obr. 5.1: Diagram aktivit, znázorňující postup práce s programem.

Na ilustraci (Obr. 5.1) je znázorněn průběh práce s pluginem, jak by mohl vypadat v reálné implementaci. Počátek uživatelských akcí začíná v momentě, kdy je vybrána kamera. Systém zkontroluje, zda se jedná o validní instanci kamery a v kladném případě ji zduplikuje a kopii označí jako referenční. Tímto okamžikem počínaje a volbou renderu konče, nezáleží na pořadí kroků, ve kterých jsou jednotlivé akce vykonávány. Například je možné

zvolit způsob projekce, poté specifikovat odchozí soubor a menu časový průběh spolu s výstupní velikostí ponechat na výchozích hodnotách. Poznámka *sub-activity*, uvedená na diagramu aktivit průběhu práce, se týká specifitějšího nastavení, které konkrétní položka nabízí. Po vyplnění nutných políček má uživatel možnost renderovat.

#### 5.4 Uživatelské rozhraní

Na (Obr. 5.2) je k dispozici naskicovaná podoba grafického uživatelského rozhraní. Pro snazší orientaci je hlavní plovoucí okno rozděleno celkem do devíti rozbalovacích nabídek. Hierarchicky nejviditelnějším prvkem je tlačítko výběru referenční kamery. Analogicky možnost spuštění renderu se nachází ve spodní části modulu. Pořadí jednotlivých nabídek odpovídá ideálnímu pracovnímu postupu směrem shora dolů. Každá rozbalovací nabídka je opatřena krátkým a výstižným popiskem podle toho, jaké funkce nabízí.

### Camera Picker

Pick a camera

---

#### Method of Projection

CAVE  
 Autostereoscopic Display

---

#### CAVE Settings

Stereo safe frame  
 Front    Right    Top  
 Rear    Left    Bottom  
 Parallel cameras  
 Converging cameras  
 Recommended camera dist.  
 User specified camera dist.  
 Distance:

---

#### Autostereo display

Nr. of cameras:   
 Recommended camera dist.  
 User specified camera dist.  
 Distance:

---

#### Image composition

Separate files  
 Horizontally composed  
 Vertically composed  
 Swap images

---

#### Time output

Single frame  
 Active time segment  
 Range  
 From  To

---

#### Output size

Width:   
 Height:

---

#### Output folder

Folder name

---

#### Render

Prefix:   
 File extension:

Preview
Render

Obr. 5.2: Návrh grafického uživatelského rozhraní, před jeho implementací.



## 6 Implementace

Po návrhu přichází na řadu samotná implementace. V této kapitole nejprve nastíním programovací možnosti, které mi zvolený jazyk nabízí. Popíši postup při umístění jednotlivých kamer v prostoru a zmíním slepé uličky a problémy, které se nepodařilo vyřešit.

### 6.1 Vybrané funkce MaxScriptu

Jeden z důvodů, proč byl v analytické části zvolen MaxScript, je i ten, že nabízí množství již předpřipravených funkcí. V následujících odstavcích rozeberu použité funkce MaxScriptu, které umožnily vzniknout implementaci.

- `in coordsys <node>` - funkce, umožňující polohovat či rotovat objekty v souřadném systému zvolené reference. Tato funkce byla využívána především při správném pozicování nově utvářené kamery v souřadném systému referenční kamery;
- `degToRad <angle>` / `radToDeg <angle>` - zvolený úhel ve stupních je převeden na radiány pomocí funkce `degToRad`;
- `maxOps.CloneNodes <node_array>` - funkce duplikující obsah pole, poskytnutého formou argumentu. Zajímavostí této funkce je především fakt, že pokud jako argument uvedu samotnou kameru, která má přiřazen vlastní cíl, je duplikován i cíl, takže výstupem je pole o dvou prvcích;
- `getPixels <bitmap> <coord_point2> <num_pixels>` - tato funkce operuje nad načtenou bitmapou a podle délky výběru je alokováno pole, do kterého je zkopírován řádek pixelů. 2D informace počátku výběru je specifikována jako druhý argument. Analogicky funguje i funkce `setPixels`, která místo třetího argumentu definuje paletu nově určených pixelů;
- `paramWire.connect <value>from <value>to <string>toExpr` - způsob, jak přiřadit animační controller zvolenému objektu. Nejprve zvolím sledovaný parametr zdrojového objektu, poté parametr cílového objektu a definujeme funkční závislost;
- `doesFileExist` - tato funkce je využita v momentě, kdy se uživatel pokouší na disk zapsat nový soubor se stejným názvem, jako již existující.

## 6.2 *Napozicování odvozených kamer*

Postup tvorby odvozené kamery pro CAVE na základě referenční je následující:

- Získání transformační matice referenční kamery a její následná aplikace na nově vytvořenou kameru, umístěnou na souřadnici  $[0, 0, 0]$  s nulovou rotací.
- Nově vytvářená kamera je nastavena jako potomek referenční.
- Na základě budoucího umístění je kamera v referenčním systému přesunuta do takové pozice, aby snímaná plocha korespondovala s plochou referenční kamery a úhel pohledu nově pozicované kamery byl  $90^\circ$ .
- Tato kamera je zduplikována, originál je přemístěn nalevo a kopie napravo.
- Je vytvořen animační controller, který zaznamenává vzdálenost mezi referenční kamerou a jejím cílem.
- Každá z dvojice kamer je propojena controllerem, který na základě aktuální vzdálenosti referenční kamery od cíle přepočítá jejich vzájemnou vzdálenost.
- Stejný controller dopočítává i případnou změnu úhlu pohledu kamery.
- Druhý controller hlídá rotaci kamery v lokální ose  $z$  a případná změna je ihned odražena na rotaci odvozených kamer.

Tvorba odvozené kamery pro autostereoskopický displej:

- Stejně jako v případě kamery pro CAVE je i zde transformační matice referenční kamery aplikována na nově vzniklou kameru na souřadnicích  $[0, 0, 0]$ .
- Referenční kamera je rodičem nově utvářené kamery.
- Podle uživatelem specifikované vzdálenosti je kamera přemístěna do adekvátní pozice.
- Rotace referenční kamery v lokální ose  $z$  je za pomoci controlleru převáděna na stejnou rotaci kamery nové

## 6.3 *Slepé uličky*

První verze pluginu spočívala v přesném napozicování nově vzniklých kamer pouze za pomoci trigonometrických výpočtů. Postupně bylo dosaženo naprosté nezávislosti na poloze a případné rotaci referenční kamery. V průběhu práce bylo nutné začít zaměřovat se



na render animací. Původní myšlenkou bylo hierarchicky podřídit nově vzniklé kamery té referenční, přičemž výchozí kamera by řídila transformační operace. Jak se ukázalo, stereo kamery se sice pohybovaly po trase reference, ale jejich orientace v prostoru se nijak neměnila, protože údaje o jejich poloze a rotaci byly spočítány jen pro první snímek. Další možností realizace bylo vytvořit *listener*, který by zachytil událost změny rámce a nově vzniklé kamery by přemístil do odpovídající pozice. Naneštěstí jediné akce, na které je 3ds Max schopen reagovat jsou spuštění a ukončení animace a změna délky trvání. Toto zjištění vedlo k hlubšímu zkoumání možností animačních controllerů, které se o aktualizaci při přechodu na jiný snímek starají automaticky.

Užitím animačních controllerů odpadla starost o změny pozice v průběhu animace, přibyla však starost s neočekávanou rotací kamery ve vlastní ose  $z$ . Během procesu přiřazení cíle je totiž kameře zcela automaticky nastaven controller *look\_at*. Ten má na starost registrovat změny pozice cíle a na jejich základě měnit pozici kamery tak, aby *target* neustále sledovala. Bohužel je výše uvedený controller implementován poněkud nešťastně. Kamera, která má nastavenou rotaci ve vlastní ose  $z$  o tuto přijde ve chvíli, kdy je změněna poloha cíle ve vertikálním směru. V tu chvíli donutí *look\_at* změnit rotaci ve vlastní ose  $z$  na 0 a to i v případě, že dané kameře je přiřazen controller další, nastavující zmíněnou rotaci na konstantní hodnotu. S tímto typem problému se v běžné praxi animátora neseťkáte. Jedná se o specializovaný případ, který by v běžném používání ovlivnil jinak zcela korektní fungování. Pro potřeby renderovacího pluginu by bylo zapotřebí implementovat controller vlastní, který ale z důvodů velké časové náročnosti nebylo možno realizovat.



## 7 Testování

### 7.1 Testování uživatelského rozhraní

#### 7.1.1 Profil testujícího účastníka

- Vysokoškolský student, učitel, architekt, stavař, teoretický fyzik...
- Technicky vzdělaný člověk, logické uvažování, podprůměrné znalosti zpracování obrazu a následného převedení do stereo formy
- Znalost práce ve 3D editoru, alespoň základy práce s kamerou, animace

#### 7.1.2 Podmínky testování

Testování probíhá v domácím prostředí účastníka za přítomnosti zadavatele. K dispozici je osobní počítač PC se standardními vstupními a výstupními zařízeními a USB klíčenka s aktuální verzí pluginu. Jako platforma je poskytnut 64bitový operační systém Windows 7 a samozřejmě 3ds Max, konkrétně v EDU verzi distribuce 2010. Dále bude účastníkovi poskytnuta papírová verze testů a základní psací potřeby. Délka testu není časově omezena, ale doba strávená nad jednotlivými úlohami, je zaznamenávána.

#### 7.1.3 Scénář

1. Spusťte 3ds Max a vložte do jednotky USB disk. Z hlavního ovládacího panelu 3dsMax vyberte MAXScript -> Run Script... . V nově vytvořeném okně specifikujte cestu k souboru *caveRenderer.ms* na vloženém USB disku a volbu potvrďte.
2. V hlavním ovládacím panelu 3ds Max zvolte File -> Open... a vyberte soubor *scena.max* z vloženého USB disku.
3. Přemístěte modul *CAVE Renderer* na takové místo, aby Vám nebránil ve výhledu na scénu. Z menu *CAVE Renderer* vyberte možnost *Pick a Camera* v roletovém menu *Camera Picker*. Následně zvolte ve scéně kameru *Camera01*. Při správném postupu se vedle tlačítka *Pick a Camera* objeví název *Camera01*.
4. Rozkliknutím roletové nabídky *Projection Method* ověřte, zda výstupním zařízením je *CAVE*.
5. Pokračujte menu *CAVE Settings*, kde nastavíte "Zobrazovat stereo bezpečnostní rámeček" (*Show Stereo Safe Frame*), pohled zvolíte na "přední" (*front*) a "levý" (*left*).

- Zbytek parametrů v menu *CAVE Settings* ponecháte na výchozích hodnotách.
6. V menu *Image Composition* vyberete "Vertikální skládání" (*Vertically Composed*) a zaškrtnete možnost "Převrátit obrazy" (*Swap Images*).
  7. Nyní přejděte do menu *Output Size*, "šířku" (*width*) nastavte na 1024 a "výšku" (*height*) na 768. V roletové nabídce *Output Folder* klikněte na jediné tlačítko "Jméno složky" (*Folder Name*), v nově vzniklém okně vyberte umístění USB disku a pomocí tlačítka "Vytvořit novou složku" (*Create new folder*) vytvořte v daném umístění složku Test.
  8. V pořadí posledním menu *Render* vepište do kolonky "Jmenná předpona" (*Name Prefix*) předložku "test". Dále nastavte "PNG" jako výstupní formát souboru. Kliknutím na tlačítko "Náhled rozmístění kamer" (*Preview Cam Setup*) zobrazte pozice kamer.
  9. Vraťte se do rozbalovací nabídky *CAVE Settings* a v části "Vzdálenost kamer" (*Camera Distance*) přepněte na volbu "Definováno uživatelem" (*User-defined*) a jako "Vzájemnou vzdálenost" (*Relative Distance*) uveďte hodnotu 1,5. Přešuněte se do menu *Render* a klikněte na stejnojmenné tlačítko. Vyčkejte, než Vám program oznámí úspěšný konec operace.
  10. V nabídce *Image Composition* zvolte "Oddělené soubory" (*Separate Files*) a v menu *Time Output* přepněte do volby "Rozsah" (*Range*) a jako meze stanovte 15 a 20. V nabídce *Render* zvolte "Jmennou předponu" (*Name Prefix*) jako "testAnim" a tlačítkem *Render* úsek vyrenderujte.
  11. V menu *Projection Method* přepněte na "Autostereoskopický displej" (*Autostereoscopic Display*).
  12. Předchodem do nabídky *Autostereoscopic Display* zvolíte "Počet kamer" (*Number Of Cameras*) na 7, necháte přepínač v poloze "Rovnoběžné" (*Parallel*) a "Vzájemnou vzdálenost" (*Relative Distance*) upravíte na 3,2.
  13. V nabídce *Time Output* přepnete na možnost "Jednotlivý rámeček" (*Single Frame*). V menu *Render* zadejte jako "Jmennou předponu" (*Name prefix*) "autostereo". Kliknutím na tlačítko *Render* spustíte výpočet.

#### 7.1.4 Výsledky

Následující tabulka ilustruje, jak se reálný uživatel potýkal s úlohami. Relativně dlouhý čas strávený nad některými úlohami je způsoben především samotným renderem. Testující

subjekt měl rovněž prostor komentovat své akce a podnítit tak optimalizaci uživatelského rozhraní.

	1	2	3	4	5	6	7	8	9	10	11	12	13
<b>Čas</b>	53	22	41	17	35	18	76	30	74	92	8	42	70
<b>Obtížnost</b>	2	1	2	1	1	1	3	1	1	2	1	2	1

Tabulka 1: Zobrazení výsledků hodnocení, vzniklých po testování uživatelského rozhraní reálným uživatelem. 1. řádek znázorňuje pořadí úkolu ve scénáři. Zobrazené časové údaje jsou ve vteřinách, obtížnost 1-5 (5-nejtěžší)

Orientace uživatele v programu byla dostačující. Jediný vážnější nedostatek byla nejasná struktura pojmenování roletové nabídky u autostereoskopického displeje. Problém byl ve stejném pojmenování přepínače v *Projection Method* a nadpisu roletové nabídky *Autostereoscopic Display*. Na základě námětu uživatele byl za název roletkového menu přidán řetězec *Settings*.

## 7.2 Testování funkčnosti

Testování probíhalo v prostorách IIM (Institut Intermédií). K dispozici byl počítač vybavený programem Autodesk 3ds Max v 64bitovém vydání. Pro testovací účely byla vytvořena scéna města, tvořená budovami, vygenerovanými na základě skriptu Building Generator (<http://www.tysonibele.com/Main/BuildingGenerator/buildingGen.htm>).

První rendery vznikaly jako statické a sloužily pro určení správné vzájemné vzdálenosti stereo kamer. Následně byly snímky užitím pasivní stereoskopické projekce promítnuty a na základě subjektivního dojmu byla výsledná vzájemná vzdálenost stanovena na 5 jednotek.

Animace vznikla jako jednoduchý průlet výše zmíněným městem, kdy se kamera spolu s cílem pohybovala po křivce. Spuštění skriptu si vyžádalo zásah do kódu, neboť se projevila chyba špatného výstupního formátu. Přestože bylo nastaveno generování animace, po skončení výpočtu byl výsledným souborem obrázků ve formátu JPEG. Po zprovoznění renderu byla výstupem krátká sekvence, otestovaná užitím pasivní projekce a shledána jako vyhovující. Jediným nedostatkem projekce byl trhaný průběh přehrávání, způsobený vyšším rozlišením videa a menším výkonem počítače.



## **8 Závěr**

### ***8.1 Zhodnocení dosažených cílů***

Podářilo se realizovat zásuvný modul pro 3ds Max, generující novou sadu kamer pro CAVE a autostereoskopické displeje. Vzniklé kamery jsou uspořádány tak, aby dokázaly nejen snímat statický snímek, ale i korektně zaznamenávat snímané okolí v průběhu animace. Jsou implementovány i další obsluhující události, blíže související s renderem. Program je ovládán přes intuitivní grafické uživatelské rozhraní. Na základě obtíží specifikovaných v implementační části, nebylo možné vytvořit horní a dolní pohled pro CAVE. Výsledná funkčnost pluginu byla otestována v Institutu intermédií.

### ***8.2 Doporučení dalšího pokračování práce***

Možnou návazností této práce by mohlo být např. odstranění problémů spojených s užitím animačních controllerů. Konkrétně implementaci controllerů vlastních, které se budou chovat korektně při libovolném narotování ovládaného objektu. Ke zlepšení uživatelského komfortu by přispělo i automatické aktualizování rozmístění kamer ihned po změně klíčových parametrů. Dále přichází v úvahu možnost prokládaného ukládání stereoskopické animace v komprimované i nekomprimované podobě.





## 9 Použité zdroje

- [1] John P. Costella, *A Beginner's Guide to the Human Field of View*, The University of Melbourne, School of Physics, 1995
- [2] Whitman Richards, D. Regan, *A stereo field map with implications for disparity processing*, Investigative Ophthalmology, Str. 904 - 909, 1973
- [3] Petr Čížek, *Prostorové zobrazování*, Diploma Thesis, Západočeská univerzita v Plzni, 2005
- [4] Kirill Dmitriev, Thomas Annen, Grzegorz Krawczyk, Karol Myszkowski, Hans-Peter Seidel, *A CAVE System for Interactive Modeling of Global Illumination in Car Interior*, MPI Informatik, 2004
- [5] Richard Bayer, *Stereoskopický výstup z programu 3ds Max*, Bachelor Thesis, Masarykova univerzita, 2007
- [6] Daniel Laubr, *Stereoskopická projekce*, Diploma Thesis, ČVUT Praha, 2006
- [7] Roman Berka, *Stereoskopie*, přednáška MM1, 2010
- [8] Autodesk, *3ds Max 2009 User Reference*, 2008
- [9] VRCO Inc., *CAVELib, User and Reference Guide*, 2004
- [10] Graham Jones, Delman Lee, Nicolas Holliman, David Ezra, *Controlling Perceived Depth in Stereoscopic Images*, Sharp Laboratories of Europe Ltd., [2000]
- [11] Achille Peternier, Sylvain Cardin, Frédéric Vexo, Daniel Thalmann, *Practical Design and Implementation of a CAVE System, High Quality Four Walls CAVE howto*, [2006]
- [12] Autodesk, *3ds Max 2009 MaxScript Reference*, 2008
- [13] Samuel P. McMullen, *Stereo 3D: A Study on Urgency, Agency and Realism & Their Effect on Video Game Immersion*, 2004
- [14] Helmut Jorke, Markus Fritz, *Infitec - A New Stereoscopic Visualisation Tool By Wavelength Multiplex Imaging*, INFITEC GmbH, [2005]
- [15] D. Bowman, D. Raja, *A method for quantifying the benefits of immersion using the CAVE*, 2004
- [16] Wikipedia - Stereoscopy  
<http://en.wikipedia.org/wiki/Stereoscopy>. K datu 14.5.2010
- [17] Wikipedia - Binocular disparity

[http://en.wikipedia.org/wiki/Binocular\\_disparity](http://en.wikipedia.org/wiki/Binocular_disparity). K datu 14.5.2010

[18] Wikipedia - CAVE

[http://en.wikipedia.org/wiki/Cave\\_Automatic\\_Virtual\\_Environment](http://en.wikipedia.org/wiki/Cave_Automatic_Virtual_Environment). K datu 14.5.2010

[19] Wikipedia - Spatial perception

[http://en.wikipedia.org/wiki/Depth\\_perception](http://en.wikipedia.org/wiki/Depth_perception). K datu 14.5.2010

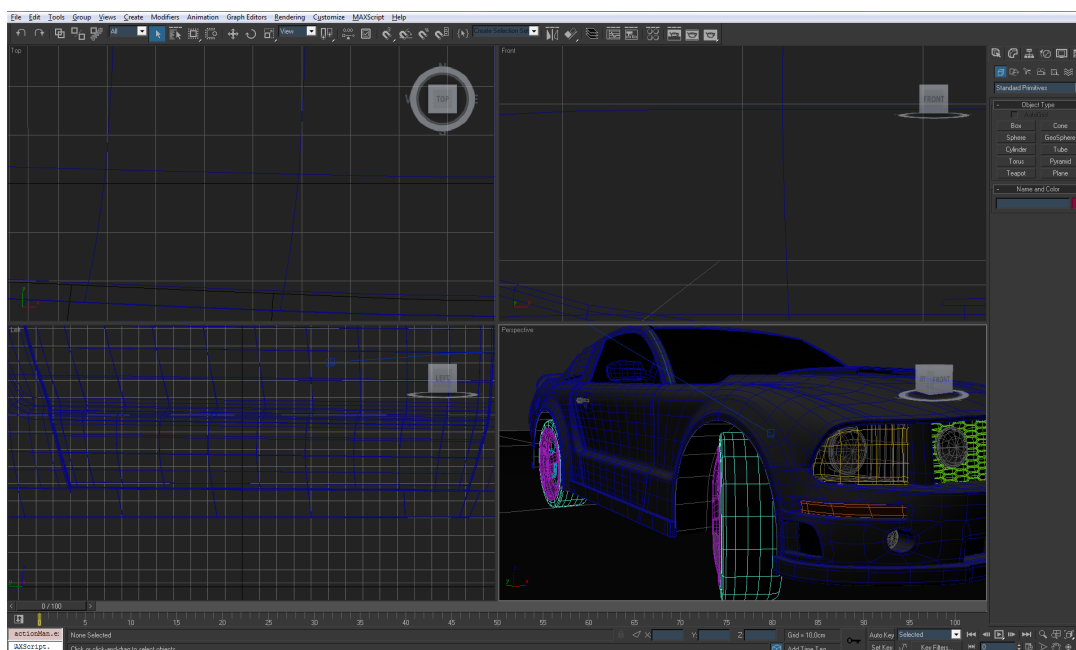
[20] Wikipedia - Accomodation

[http://en.wikipedia.org/wiki/Accommodation\\_%28eye%29](http://en.wikipedia.org/wiki/Accommodation_%28eye%29). K datu 14.5.2010

## 10 Obrazová příloha



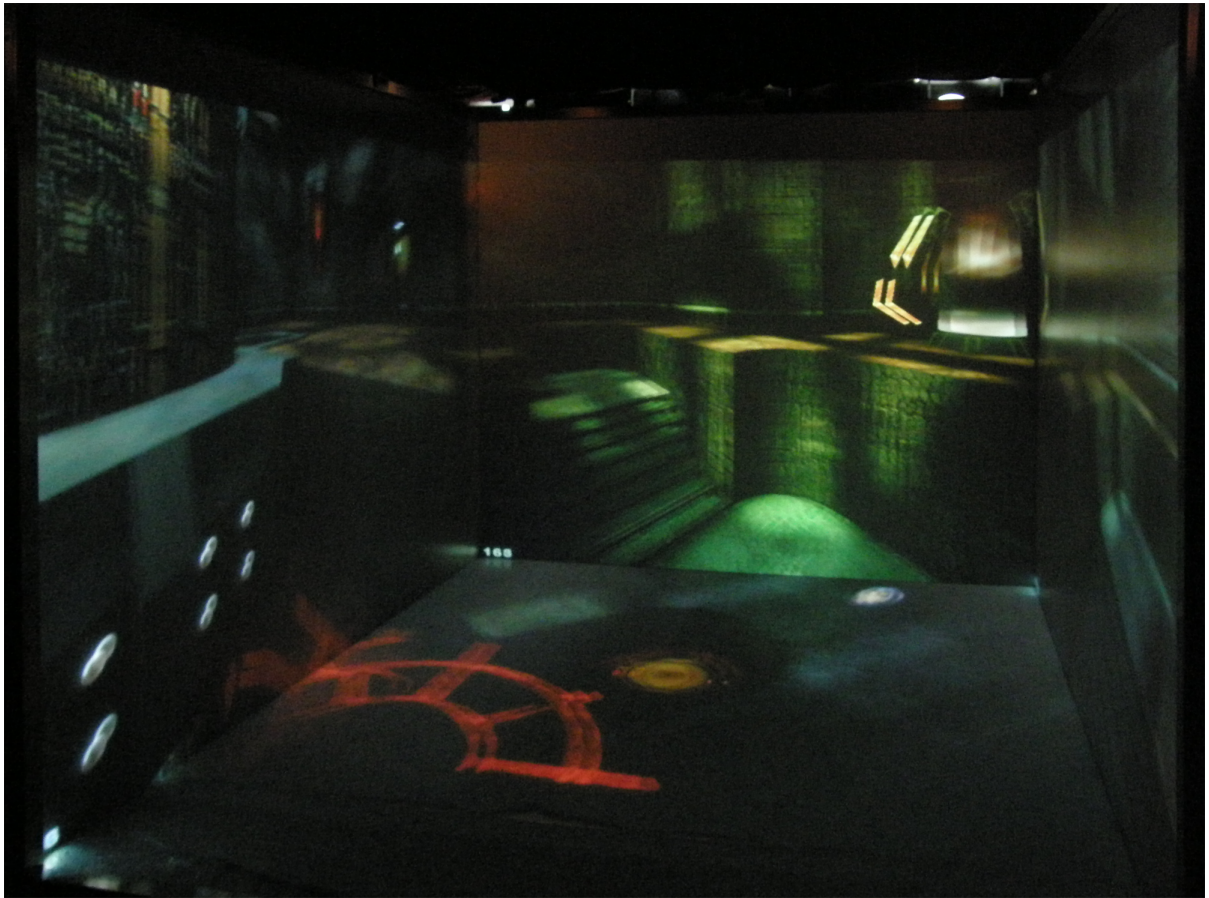
Obr. 10.1: CAVE používaný v designovém centru firmy Mercedes-Benz, divize třídy S.



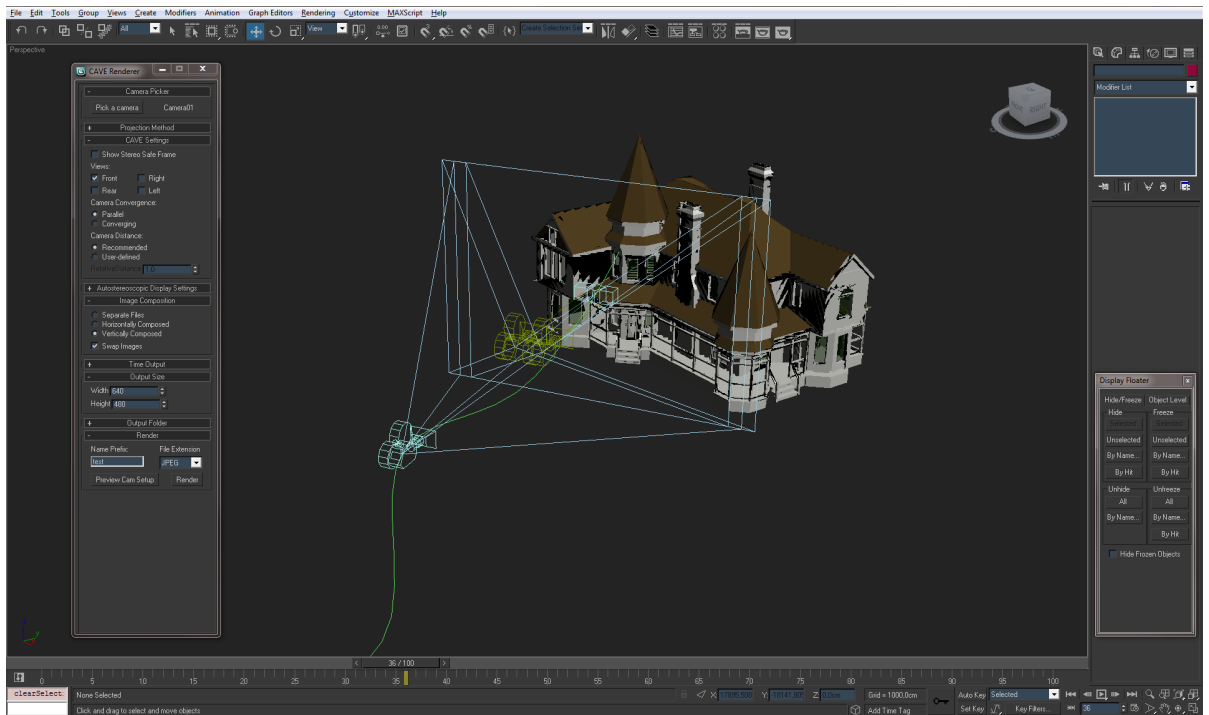
Obr. 10.2: Pohled na uživatelské rozhraní programu 3ds Max 2009.



*Obr. 10.3: Projektory pasivní stereoskopické prezentace v IIM. Sloužily k testovacím účelům.*



Obr. 10.4: Pohled na CAVE instalovaný v Institutu Intermédii na FEL ČVUT.



Obr. 10.5: Scéna s originální kamerou a na jejím základě vytvořenými stereo kamerami.



## **Dodatek A - Obsah přiloženého CD**

Součástí této práce je přiložené CD. Na následujících řádcích naleznete výpis jednotlivých adresářů spolu se stručným popisem.

### Bachelor\_Thesis

*Images - Obrázky a ilustrace, použité v bakalářské práci*

*Text - Vlastní text bakalářské práce. Formáty PDF a ODT*

### CAVE\_Renderer

*Design - Vybrané UML diagramy vzniklé během návrhu*

*Manual - Uživatelský manuál práce s pluginem*

*Plugin - Vlastní plugin CAVE Renderer*

